VSWR (Voltage Standing Wave Ratio) Meter
Technical Report

SD 0809

Aakash Malik
Sandeep Pal Singh

May 11, 2009

Table of contents

Introduction

Standing Wave:
A **standing wave** in a transmission line is a wave in which the distribution of current, voltage, or field strength is formed by the superposition of two waves of the same frequency propagating in opposite directions. The effect is a series of nodes (zero displacement) and anti-nodes (maximum displacement) at fixed points along the transmission line. Such a standing wave may be formed when a wave is transmitted into one end of a transmission line and is reflected from the other end by an impedance mismatch, i.e., discontinuity, such as an open circuit, a short or other load mismatch.

Standing Wave Ratio:

In telecommunications, **standing wave ratio** (**SWR**) is the ratio of the amplitude of a partial standing wave at antinodes (maximum) to the amplitude at an adjacent node (minimum), in an electrical transmission line.

The SWR is usually defined as a voltage ratio called the **VSWR**, for voltage standing wave ratio. For example, the VSWR value 1.2:1 denotes maximum standing wave amplitude that is 1.2 times greater than the minimum standing wave value. It is also possible to define the SWR in terms of current, resulting in the ISWR, which has the same numerical value. The power standing wave ratio (PSWR) is defined as the square of the VSWR.

Why Bother about SWR??
Knowledge of the SWR can be important because the RF losses in a transmission line increase as the SWR increases. This is especially true in coaxial transmission lines, which are generally not used when the SWR will be above 3.0. Parallel conductor ladder line, whose dielectric is primarily air, can be used at SWR's of as high as 12.0 without problems.

The most common case for measuring and examining SWR is when installing and tuning transmitting antennas. When a transmitter is connected to an antenna by a feed line, the impedance of the antenna and feed line must match exactly for maximum energy transfer from the feed line to the antenna to be possible.

When an antenna and feed line do not have matching impedances, some of the electrical energy cannot be transferred from the feed line to the antenna. Energy not transferred to the antenna is reflected back towards the transmitter. It is the interaction of these reflected waves with forward waves which causes standing wave patterns. Reflected power has two main implications in radio transmitters: The failure of the line to transfer power at the standing wave frequency will usually result in **attenuation distortion**, **Radio Frequency (RF) energy losses** increase, and **damage** to the transmitter can occur.

Objective

The primary objective of this project was to design and construct two or three devices that measure the VSWR in the frequency range 1.8 to 144 MHz, basically each of these devices have to be appropriate for the class room demonstration of the concepts of VSWR, forward power, reverse power and/or related concepts.

Requirements

Analog VSWR

- Connections to transceiver and antenna through BNC connectors
- Analog display of the forward and reverse voltages using DC micro ammeter.
- To be able to measure VSWR in terms of relative ratio of forward and reverse voltages using a toggle switch to access both modes.

Digital VSWR

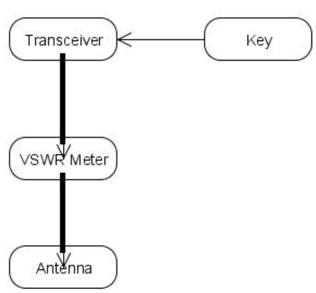
- Connections to transceiver and antenna through BNC connectors.
- To be able to measure the forward power, reverse power, actual power and VSWR directly through forward and reverse voltages.
- An LCD for displaying the forward power, reverse power, actual power and VSWR
- Power adapter 9V-12V or 12V batteries.

Advanced Requirements

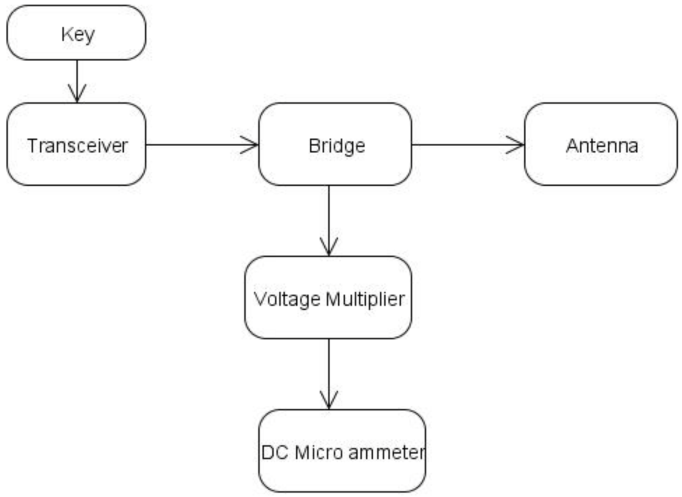
- A variable dummy load in place of antenna for testing the device at different load values.
- Transceiver or other input device with a key plugged to it.
- Turn off the LCD and sleep the micro until an input is detected by the device.

Block Diagrams

Block diagram of the entire system

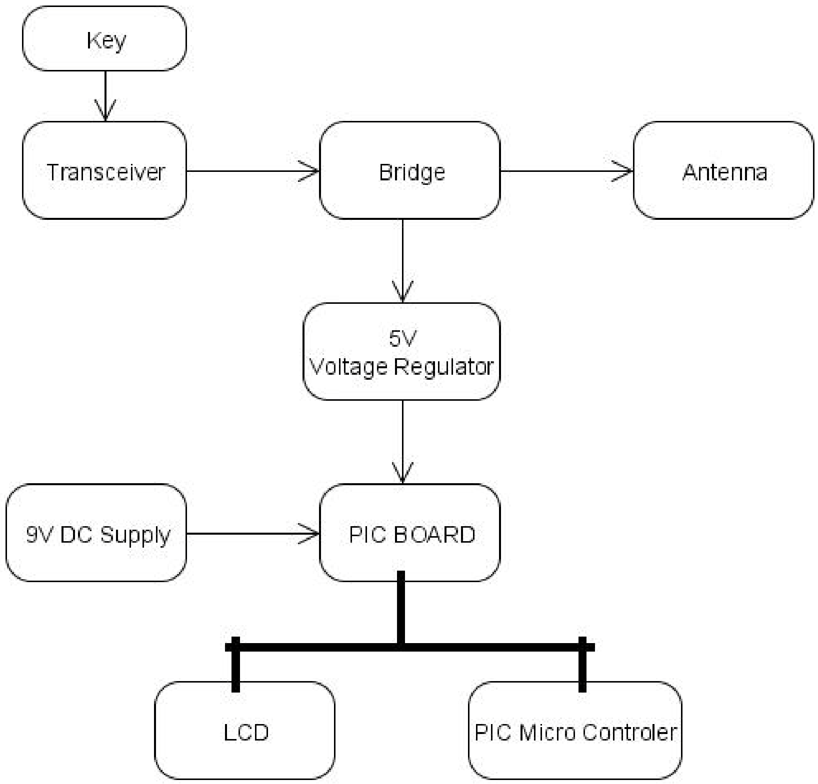


Block diagram of Analog VSWR



We used a resistance bridge circuit to have a voltage ratio relation between the forward and the reverse voltages across the two diodes, followed by a variable resistor which acts as a voltage multiplier. Finally we have a micro ammeter which reads the voltage readings.

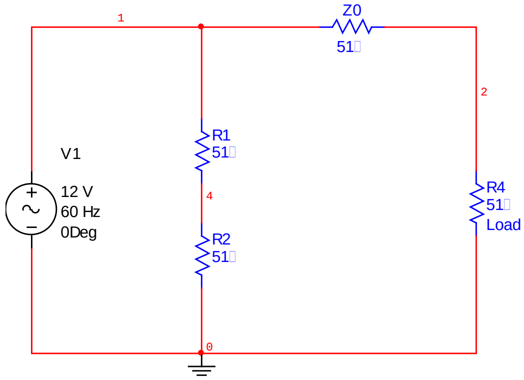
Block diagram of Digital VSWR



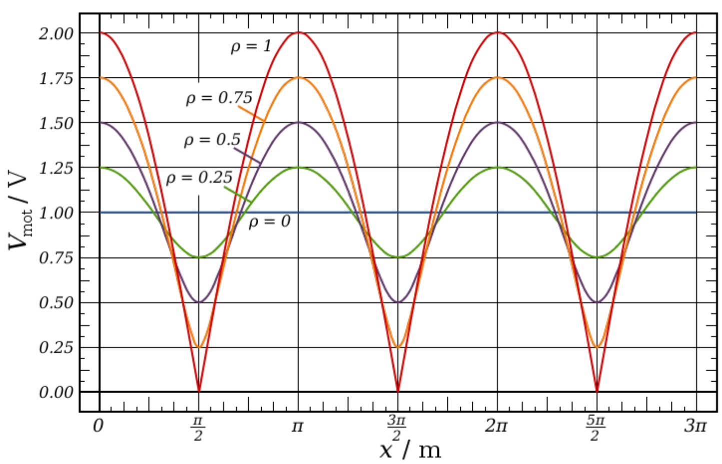
We used a 9V DC adapter as our main power supply. A DC/DC switching regulator is used to generate a 5V constant output through the bridge. This 5V voltage supply is used to power up the LCD, and the PIC Processor. Then this PIC circuit is connected to the bridge which is located between the transceiver and the antenna.

Schematics and Descriptions (Hardware)

Basic circuit



This is the basic bridge circuit that we used in our devices; the main emphasis was to use 3 50 ohm voltage divider circuits. Here we used 50 ohms because the internal load of the transceiver is itself 50 ohm so while dealing with the antenna stuff we have to be careful about matching the bridge load with that of the transceiver, otherwise it can cause damage to the transceiver. The 50 ohm load acts as an antenna. While we examined this circuit using an oscilloscope we got the following result.



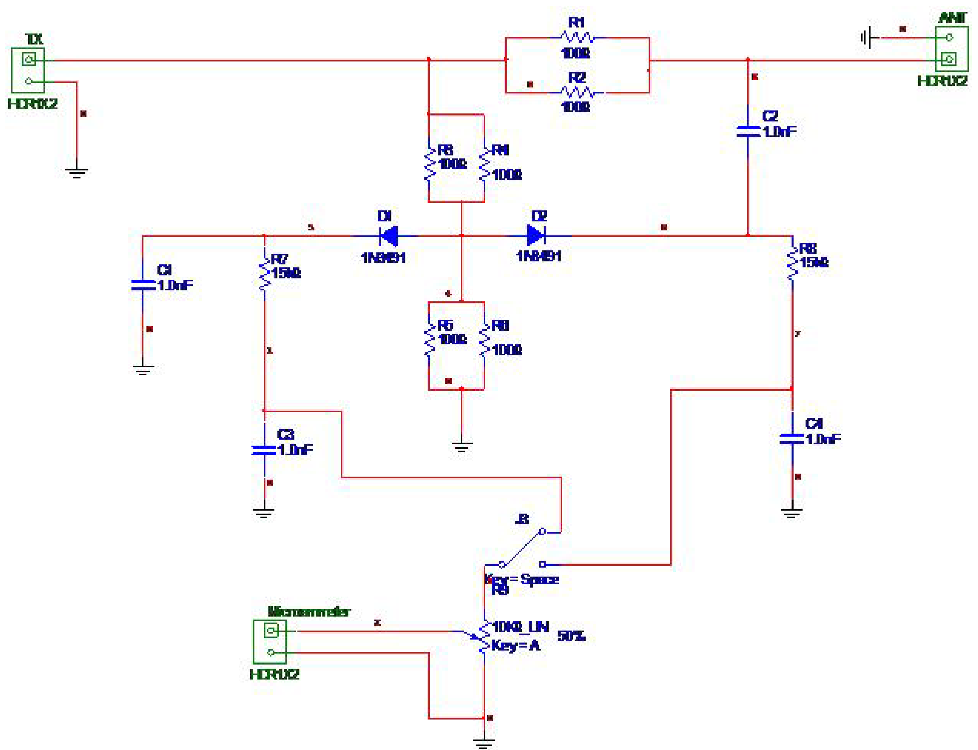
Where V_{mot} is the maximum voltage over time at a distance x from the transmitter and is the periodic function dependent on ρ. Here ρ is the magnitude of the reflection coefficient Γ.

$$\Gamma = \frac{V_r}{V_f}.$$

$$\rho = | \Gamma |$$

$$VSWR = \frac{V_{\max}}{V_{\min}} = \frac{1+\rho}{1-\rho}.$$

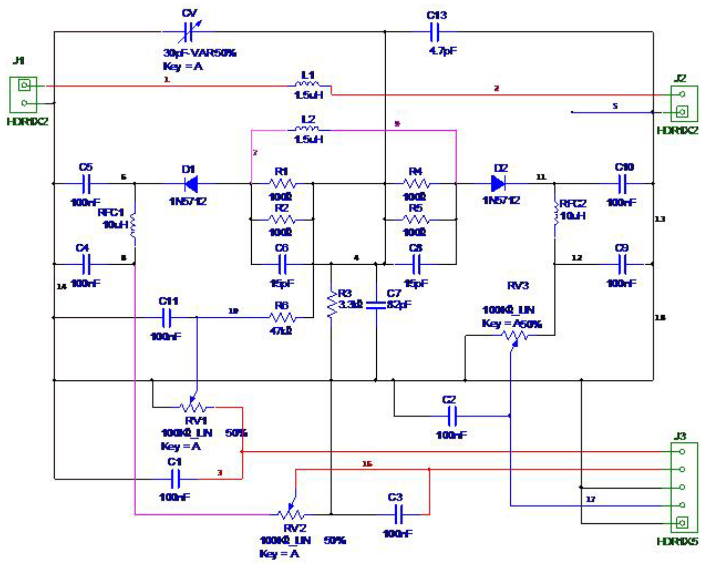
Analog VSWR Schematic



The connector TX and ANT are used for connecting the device to the transceiver and the antenna respectively. The two resistor pairs (R1-R2 & R3-R4) between TX and ANT works as the bridge circuit along with (R5-R6). All of them work as a voltage divider circuit along with the other pairs. The resistor pair R1-R2 are also in parallel with the ANT load, so this makes another voltage divider circuit which helps in calculation of reverse voltage as a ratio of forward voltage. The two diodes D1 and D2 act as the forward and reverse voltage rectifiers respectively. We used a toggle switch (J3) to toggle between forward and reverse mode, which follows to a variable resistor connected to the ammeter. Variable resistor is used to scale the meter to full scale in the forward mode. Then the meter gives the reading of voltage.

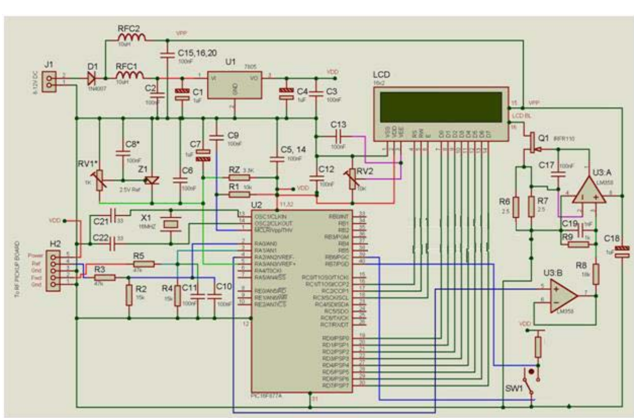
Digital VSWR Schematic

Bridge Circuit



In the above schematic, same resistance bridge circuit (R1-R2 & R4-R5) followed by the rectifying diodes (D1 & D2) is used but with some modification to get a more advanced device. The connectors TX and ANT are used to connect the device to the transceiver and the antenna respectively. Then we used a torrid (L1) in between the connectors so that we can get rid of the radio frequency interferences. The capacitor-inductor combination used in front of both the diodes works as a filtering circuit and helps us to get rid of the main hums, it also helps to act as a switching circuit and gives the ratio between the input and output voltages across it as an inductor energizes for a definite fraction of frequency and de-energizes for rest of it. This voltage is then read across the variable resistor (RV2 & RV3). The resistance RV1 provides logic 5V for deriving the PIC and LCD, its value is adjusted at the time of calibration. Finally, the forward voltage, reverse voltage and power (through RV1) are connected to the H1 port which goes to PIC board for calculations of power and VSWR as well as for the analog to digital conversion.

PIC Board



The PIC board takes the values of RV1, RV2, V_{fwd}, V_{rev}. Then the PIC converts them to the digital and then the calculations are made to get the values of forward power FWD, reverse power REV, net power P and SWR. And then the LCD is used to display the result.

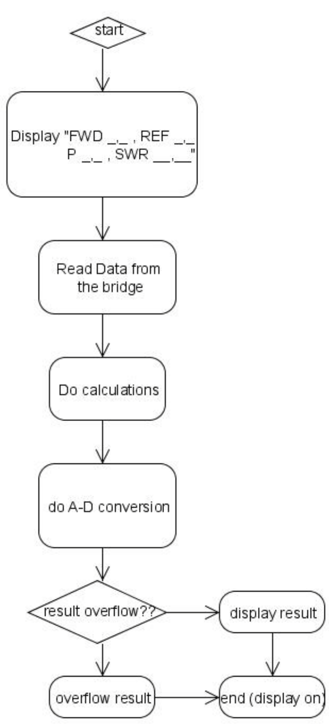
Calibration of the bridge used in Digital VSWR

After connecting the bridge with the PIC board-

- Connect the variable dummy load to the ANT connector of the bridge board.
- Connect the transceiver to the TX connector of the bridge board.
- Turn on the PIC board by 9V-12V DC adapter.
- Now by using a multi-meter adjust R1 for a reading of 0.65V.
- Set the transceiver to 3.8 MHz
- Key in the transceiver, and note the readings.
- Adjust the variable capacitor for a REF value equaling 0.
- Now replace the PIC board with some known SWR meter and set the transceiver on high power.
- Now adjust RV2 for a reading of high power at FWD and the reading at REF still 0.
- Swap the dummy load with transceiver and repeat the above two steps
- You can repeat all these steps several times to get more and more accuracy.

Software

Software flow chart



Micro-controller code

The codes for this project have been written and tested in MPLAB Integrated Development Environment

```
// Digital VSWR
// Group SD0809

// Global Variables
unsigned int A2D, A2Dact1, A2Dact2, FWD, REV,
P, SWR;
const unsigned char MSG0[16] = "FWD _,_ REV
_,_
P _,_ SWR _,_ ";

// Subroutine Declarations
#include <pic.h>

static volatile unsigned int TIMER1 @ 0x0E;
static volatile unsigned int CAPTURE1 @ 0x15;
static volatile unsigned int CAPTUER2 @ 0x1B;

#include        "Lcd_16x4.h"
#include        "FUNCTION.h"
#include        "A2D.h"
void Init_Port(void);
void Init_PWM(void);
void Init_INT(void);
void LCD_Out(unsigned int DATA);

// Subroutines
#include        "Lcd_16x4.c"
#include        "FUNCTION.c"
#include        "Bootloader.c"
#include        "A2D.c"

void Init_Port(void)
{
TRISA = 0x0F;
TRISB = 0x03;
TRISC = 0;
PORTB = 0;
PORTC = 0;
}
void Init_PWM(void)
{
PR2 = 77 ;
T2CON = 0x07 ;
CCPR1L = 0x26 ;
CCP1CON = 0x3C ;
PEIE = 1;
TMR2ON = 0;
}
// Main Routine
void main(void)
```

```
void Init_INT(void)
{

INTEDG = 1; // interrupt on a falling edge
INTE = 1;
INTF = 0;
}

void LCD_Out(unsigned int DATA)
{
  unsigned char A[5], i;
  for (i=0; i<5; i++) {
    A[i] = DATA % 10;
    DATA = DATA / 10;
    }
  lcd_write(ascii(A[4]));
  lcd_write(ascii(A[3]));
  lcd_write('.');
  lcd_write(ascii(A[2]));
}

// Interrupt service routine
void interrupt IntServe(void) @ 0x10
{
 if (INTF) {
                        A2Dact1 = A2D_Read1();
                        A2Dact2 = A2D_Read2();
                        CCP1CON = 0x2C;
                        TMR2ON = 1;
                        RC1 = 1;
                        RC3 = 1;
                        FWD = ((A2Dact1)^2)/RB1;
                        REV= ((A2Dact2)^2)/RB2;
                        P = FWD – REV;
                        SWR=
(A2Dact1+A2Dact2)/(A2Dact1-A2Dact2);
          INTF = 0;
          }

}



if (RC6 == 0){

          lcd_move(3,0);

          for (i=0; i<18; i++)
                    lcd_write(MSG2[i]);

          RC6 = 1;

          }
```

```
{
  unsigned char i;
  unsigned char A[5];


        Init_Port();
        Init_PWM();
        Init_INT();
        lcd_init();
        A2D_Init();
        Wait_ms(100);
   A2D = 0;
        A2Dact = 0;
        Sleeptimer = 0;
        Wait_ms(100);
        lcd_move(0,0);
        for (i=0; i<16; i++) lcd_write(MSG0[i]);


        GIE     = 1;
        while (1){
                if (RC7 == 1){
                        lcd_inst(0x0C);
                        }
                if (RC4 == 0){
                        lcd_move(0,0);
                        for (i=0; i<16; i++)
lcd_write(MSG0[i]);
                        RC4 = 1;
                        }

                A2D = A2D_Read1();
                Force = 31*A2D - 3125;

                if (A2D > 90){

                        if (RC5 == 0){
                                lcd_move(2,0);
                                for (i=0; i<13;
i++) lcd_write(MSG1[i]);
                                RC5 = 1;
                                }
                        lcd_move(2,5);
                        LCD_Out(Force);
                        } else {
                                lcd_move(2,0);
                                for (i=0; i<13;
i++) lcd_write(MSG4[i]);
                                lcd_move(3,0);
                                for (i=0; i<18;
i++) lcd_write(MSG5[i]);
                                RC5 = 0;
                                RC6 = 0;
                                }
```

Other codes used in cohesion with the above code

A2D.c

```
// A/D Routines

void A2D_Init(void)
{   ADCON0 = 0x041;
    TRISA  = 0x0F;
    ADCON1 = 0x80;
    }

unsigned int A2D_Read1(void)
{
  unsigned int result;
  ADCON0 = 0x81;
  ADGO = 1;
  while (ADGO);
  result = (ADRESH << 8) + ADRESL;
  return(result);
}

unsigned int A2D_Read2(void)
{
  unsigned int result;
  ADCON0 = 0x81;
  ADGO = 1;
  while (ADGO);
  result = (ADRESH << 8) + ADRESL;
  return(result);
}
```

Bootloader.c

```
// bootloader routine

#asm
  org   0x0000
  clrf  3
  movwf 0
  movlw 0x0A
  goto  start
  #endasm
```

Function.c

```
// ---  FUNCTION.C --------------
//  Generally useful functions
//
//  void hex2dec(unsigned int X, unsigned char *A[5])
//    separate an unsigned integer into decimal
//         A[4] * 10E4
//         A[3] * 10E3
//         A[0] * 1
//
// void Wait_ms(unsighed int X)
//    wait approximately X milliseconds and return

void hex2dec(unsigned int X, unsigned char *A[5])
{
  unsigned int Y;
  unsigned char a4, a3, a2, a1, a0;
  unsigned char i;

  for (i=0; i<5; i++) {
    A[i] = X % 10;
    X = X / 10;
    }
  }
// ---- WAIT_MS ----------------
void Wait_ms(unsigned int X)
{
  unsigned int i, j;

  for (i=0; i<X; i++) {
    for (j=0; j<500; j++);
    }
  }
// - ASCII ------------------------

unsigned char ascii(unsigned char c)
{
  c = c & 0x0F;
  if (c < 10) return (c+48);
```

LCD_20x4.c

```
// --------- LCD.C ---------------------
//
// This routine has LCD driver routines
//
//  LCD_INST:   send an instruction to the LCD
//  LCD_DATA:   send data to the LCD
//  LCD_INIT:   initialize the LCD to 16x4 mode
//
//--- Revision History -----------------
//     5/20/00   JSG
//     9/27/00   Modify LCD_HELLO for Jump messages
//    11/05/00   Clean up LCD routine to use less RAM

// LCD Service Routines

static bit LCD_RS       @ ((unsigned)&PORTB*8+2);       // Register
select
static bit LCD_EN       @ ((unsigned)&PORTB*8+3);       // Enable

// #define     LCD_STROBE          ((LCD_EN = 1),(LCD_EN=0))

void lcd_pause(void)

{
  unsigned char x;
  for (x=0; x<20; x++);
  }


void lcd_strobe(void)
{
  LCD_EN = 0;
  lcd_pause();
  LCD_EN = 1;
  lcd_pause();
  LCD_EN = 0;
  lcd_pause();
  }

//  write a byte to the LCD in 4 bit mode

void lcd_inst(unsigned char c)
{
   LCD_RS = 0;       // send an instruction
           PORTB = (PORTB & 0x0F) |  (c & 0xF0);
           lcd_strobe();
           PORTB = (PORTB & 0x0F) |  ((c<<4) & 0xF0);
           lcd_strobe();
           Wait_ms(5);
}

void lcd_move(unsigned char Row, unsigned char Col)
{
  if (Row == 0) lcd_inst(0x80 + Col);
  if (Row == 1) lcd_inst(0xC0 + Col);
  if (Row == 2) lcd_inst(0x94 + Col);
  if (Row == 3) lcd_inst(0xD4 + Col);
  }

void lcd_write(unsigned char c)
{
   LCD_RS = 1;       // send data
           PORTB = (PORTB & 0x0F) |  (c & 0xF0);
           lcd_strobe();
           PORTB = (PORTB & 0x0F) |  ((c<<4)  & 0xF0);
           lcd_strobe();
           }

void lcd_init(void)

{
//   TRISB = 0xC0;       // 16x4 LCD Boards
  TRISB = 0x03;       // 20x4 LCD Boards

  lcd_inst(0x33);
  lcd_inst(0x32);
  lcd_inst(0x28);
  lcd_inst(0x0E);
  lcd_inst(0x01);
```

else return(c + 55); }	lcd_inst(0x06); }

Detailed theory of operation

- ***Analog VSWR***

The basis of working of this device lies behind the principle of voltage division and rectification. The resistor pair R1-R2 are in parallel with the resistor pair R3-R4, and the resistor pair R3-R4 are in parallel with R5-R6, so both the diode sees half of the applied voltage on their back side. Now the diode D1 always sees a different voltage on the front side so it will always rectify, but the diode D2, due to the presence of ANT will not rectify all the time. The ANT load comes in parallel with the R1-R2 resistor pair so the voltage division applies again. Say for instance we took 100 ohm of antenna; D1 will as always see half the input voltage, while D2 will have half the input voltage on one side and only two third on the other side (due to voltage division between R1-R2 & 100 ohm antenna). So it sees one sixth of the input voltage, which is one third as much as D1 sees. Now here is why we said that it calculates the VSWR in terms of relative voltage ratio. Now the working goes this way, turn the device into forward mode (FWD) using the toggle switch. Key in the transceiver and adjust the variable resistor for a full scale reading. Release the key and turn the device to reverse mode (REV). Key in the transceiver again and note the reading. Now to check the result this value should be equal to one third of the value you got in the FWD mode i.e. one third of the full scale. Now the VSWR is calculated in the following way:

- Divide the reading you got in the reverse mode with your full scale reading, and let's call it 'p'.
- Now your VSWR will be-

$$\frac{1+p}{1-p}$$

- ***Digital VSWR***

This device uses the principle of the analog meter with few modifications which will help in better performance. The device uses the same voltage divider circuit followed by the diodes, but here we introduced a combination of capacitor and inductor after each diode to get rid of the main hums and to help in the calculation of voltages. The output from the variable resistors RV1, RV2 & RV3 is sent to the PIC board for the calculation of forward power, reverse power, net power and VSWR.

First the analog values are converted to digital and then we do the calculations.

From RV2 & RV3 we directly get the forward and the reverse voltages respectively, and then we used these voltages along with the value of RV2 and RV3 to find out the forward and reverse power by using:-

$$P = I^2 \cdot R = \frac{V^2}{R}$$

So for forward power we used V_{Fwd} and RV2 in place of V and R in the above equation and we got our P_{fwd}. And similarly we got our P_{rev} by using V_{rev} and RV3.

The net power is $P = P_{fwd} - P_{rev}$

And the VSWR is just

$$VSWR = \frac{V_{\max}}{V_{\min}} = \frac{1+\rho}{1-\rho}.$$

Where,

$$V_{\max} = V_f + V_r = V_f + \rho V_f = V_f(1+\rho).$$

&

$$V_{\min} = V_f - V_r = V_f - \rho V_f = V_f(1-\rho).$$

This result is displayed on the LCD.

When the unit is powered up, the PIC initializes its ports B and C to '0'. It also sets RA0, RA1, RB0 and RB1 as input pins and all RC pins as output pins. It initializes pulse width modulation configuration variables as shown:-
Prescalar = 16, for a prescalar of 16 T2CKPS1:T2CKPS0 = 1 : x

T2CON

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	TMR2ON	T2CKPS1	T2CKPS0
	-	-	-	-	-	1	1	x

Therefore, T2CON = 0x07

Period = (PR2+1). Prescalar. Clock period
frequency = clock period/(PR2+1)Prescalar
for obtaining a frequency of 4 KHz
$4000 = 5 \times 10^6/(PR2+1).16$
PR2 = 77
TMR2ON is initialized to '0' as we want the pulse width modulation.
CCPR1L =0x26= 00100110
CCP1CON=0x3C

The PIC then initializes INT interrupt configuration variables as follows:-
INTEDG = '1' as we want it to interrupt on a falling edge of RB0
INTE ='1'for enabling RB0 interrupts
INTF='0' for clearing prior RB0 interrupts

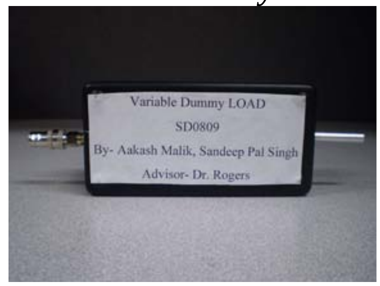
It then sets GIE to '1' to enable all interrupts and a sleep timer to '0'.
The microcontroller reads analog input from the load cell through the RA0 pin. The code uses three analog input variables A2D and A2Dact1, A2Dact2 for RV1, RV2, V_f and V_r.

The output from the sensing circuit is connected to RB0. It reads the analog inputs and calculates the corresponding values. It sets CCP1CON to 0x2C. And it displays the values.

Column1	Column2	Column3	Column4
		Final Budget	
	Parts Name		**Ammount ($)**
1	Microammeter		65
2	SPDT Toggle Switch		7.13
3	Capacitor		15
4	Resistors		10
5	Variable Resistor		4.15
6	Variable Capacitor		3.12
7	Germanium Diode IN270		7
8	coaxial Connector		6.12
9	PIC 16F877A		7.32
10	Rotary Switch 2pole 2 position		2
11	5 Coaxila Connectors		6
12	Potentiometer		3
13	Trimmer Capacitor		1
14	Inductor		4
15	1N5179 Diode		5
16	Ferrite bead		1.67
17	5K 10T Bourns Preset		2
18	100K 10T Bourns Preset		3
19	RG174 Coax Cable 7cm long		4
20	Project Enclosure I		5.76
21	Project Enclosure II		5.76
22	Project Enclosure III		3.56
23	Toggle Switch		2.134
24	LCD Display		19
	Total		192.724

Final product (pictures)

Analog VSWR ***Digital VSWR*** ***Variable Dummy Load***

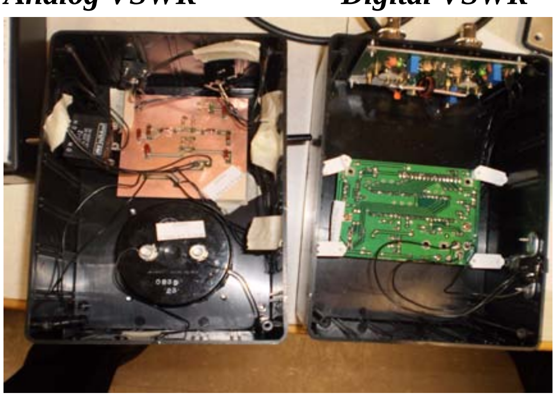






Inside View

Analog VSWR ***Digital VSWR***



Final Products

Digital & Analog VSWR with Dummy load

Technician's Troubleshooting

Analog VSWR

Meter does not response.

- Is the device getting any input?
- Check the connections.

Meter does not display correct readings.

- Check the connections
- Check if correct load is connected or not.
- If the connections are all correct then try calibrating the meter by turning the needle screw that is located right below the display.
- Check what frequency and power level you are working with.
- Make sure that you are no using old and long coaxial cable.

Digital VSWR

LCD does not display anything.

- Is the device on?
- Check the connections.

LCD does not display readings.

- Turn off the device and then turn it on again.
- Try changing the processor PIC16F877A
- Make sure that the transceiver/ input device and the antenna/ dummy load are connected properly to the Digital VSWR meter.
- Make sure the submersible force probe is connected to the touchpad testing device
- Check what frequency and power level you are working with.
- Make sure that you are no using old and long coaxial cable.

Project issues/problems/design flaws and lessons learned, future improvements

Designing the PCB and working with the PIC micro controller were the main problems that we encountered during the construction of the project. The thing that we learned about the PCB designing was that if, you are doing it for the first time then keep your traces and pads wider so that soldering will become easier and you won't damage your board. The PIC micro controller was new for us, but we had done work on other micro controller so we studied about PIC through Dr. Glower's site. Debugging the device after populating the PCB was very time consuming. This process would have been much less stressful if we had utilized better time management. We have learned that sufficient time must be allowed for debugging, because, most of the time, the design doesn't work properly the first time. In the future, this product could be improved by devising a way to power down the unit completely when not in use in order to maximize battery life. Forward and reverse voltages can also be displayed very easily.

Conclusion

Both of us learned a great deal over the course of the last two semesters. We learned the value of creating and following a timeline, communication between group members, both verbal and written, and process of designing a project from ground up. During the project, we had help along the way from our advisor, Dr. Rogers. All of the advice provided was very helpful in understanding the design process's many challenges. After the work we put in over the course of two semesters, it was very satisfying to see our design working and meeting all of the requirements.

Additional device:
Variable Dummy Load

A **dummy load** is a device used to simulate an electrical load, usually for testing purposes.

In radio this device is also known as a **dummy antenna** or a **radio frequency termination**. It is a device used in place of an antenna to aid in testing a radio transmitter. It is substituted for the antenna while adjusting the transmitter, so that the transmitter does not interfere with other radio transmitters during the adjustments. If a transmitter is tested without a load, such as an antenna or a dummy load, the transmitter could be damaged. Also, if a transmitter is adjusted without a load, it will operate differently than with the load, and the adjustments may be incorrect.

The dummy load ordinarily should be a pure resistance; the amount of resistance should be the same as the impedance of the antenna or transmission line that is used with the transmitter (usually 50 Ω or 75 Ω). The radio energy that is absorbed by the dummy load is converted to heat. A dummy load must be chosen or designed to tolerate the amount of power that can be delivered by the transmitter.

The ideal dummy load provides a standing wave ratio (SWR) of 1:1 at the given impedance.

So for the testing purpose we build a variable dummy load which has load values of 0, 22, 50, 75, 100 and ∞ ohm.

The circuit used is: